Simplification of Symbolic Network Functions using Large-Change Sensitivities

Franciszek Balik* and Benedykt Rodanski**

Abstract – **In this paper we present a brief description of a new method for simplification of symbolic network functions. Our approach is based on utilising the large-change sensitivities, both single- and multi-parameter, in order to select elements to be excluded from the formula. Our technique belongs to the simplification before generation (SBG) group of methods. The main advantage of the described methodology is the small number of network function recalculations and continuous monitoring of network function behaviour during the simplification process. Basic theoretical principles of the method and an illustrative example are presented.**

1 INTRODUCTION

In recent years symbolic analysis of electronic circuits has been experiencing a revival mainly due to increased sophistication of algorithms and a dramatic increase in computing power available to researchers and circuit designers. Several important developments have been reported in symbolic analysis of large-scale circuits. One of the more important issues is the problem of simplification of usually extremely complex expressions generated for such circuits. This complexity makes interpretation of the final formula impossible (even for medium-size circuits) and slows down the numerical calculation process.

To address this problem a number of approaches have been proposed to simplify the final formula, while maintaining the numerical accuracy above a predetermined level. Those approaches can be classified into three groups:

a) simplification before generation (SBG),
b) simplification during generation (SDG),
c) simplification after generation (SAG).

Various simplification methods have been described in [1,2] with references to the relevant original works.

A common feature of all currently published simplification (sometimes incorrectly named 'approximation') techniques is that the candidates for elimination are chosen either by a trial-and-error process or differential sensitivities are used to rank the candidate components. Differential sensitivities do not give an easy indication of the network function variations in case of removal of circuit components, when relative parameter changes vary from -100% to infinity.

Here we describe a simplification method using the large-change sensitivities to decide which components are to be removed and in what order. The method used to calculate the large-change sensitivities has been described in [3,4]. First, we calculate the single-parameter large-change sensitivities to parameter changes that make the admittance or impedance zero (component removal by either open- or short-circuiting) and sort the components in ascending order of these sensitivities. (We will use the term ***extreme-change sensitivities*** to describe this special type of sensitivities.) Next, we calculate the multi-parameter extreme-change sensitivity, iteratively adding consecutive components from the ranking list obtained earlier, until a predetermined value is reached. Components thus selected are then removed from the circuit. This reduces the complexity and the number of arithmetic operations in the final formula.

It is important to point out that our approach needs only one calculation of the network function of the full circuit. Subsequent computations use the results of this initial calculation, significantly reducing the computational effort.

2 THEORETICAL BASIS OF THE PROPOSED METHOD

Let the circuit property of interest be described by a network function $T(s,\mathbf{p})$, where: s – complex frequency, $\mathbf{p} = [p_1, p_2,\ldots, p_m]^{\mathrm{T}}$ – vector of circuit parameters. The relative change of $T(s,\mathbf{p})$, caused by simultaneous change of m circuit parameters, is given by:

$$\delta_{\mathbf{p}}^{\mathrm{T}} = \frac{T_m(s,\mathbf{p}) - T_0(s,\mathbf{p})}{T_0(s,\mathbf{p})}, \qquad (1)$$

where: T_0 – initial value of the function,
T_m – value of the function after m parameters have changed.

* Wrocław University of Technology, Institute of Telecommunication and Acoustics, ul. Janiszewskiego 7, 50-370 Wrocław, Poland. E-mail: franciszek.balik@pwr.wroc.pl
** University of Technology, Sydney (UTS), Faculty of Engineering, P.O. Box 123, Broadway, NSW 2007, Australia. E-mail: ben.rodanski@uts.edu.au

The large-change sensitivity of a network function $T(s,\mathbf{p})$ to variation of a single parameter p_i is defined as [5]:

$$\delta^T_{p_i} = (\Delta T/T)/(\Delta p_i/p_i). \tag{2}$$

If a parameter varies by 100%, i.e., $\Delta p = \pm p$, then $\Delta p/p_i = \pm 1$. Parameter (symbol) p_i can be removed from the formula if the extreme-change sensitivity and the network function satisfy the following condition:

$$\left|\delta^T_{p_i}\right| < \varepsilon \Leftrightarrow \left|\Delta T/T\right| < \varepsilon, \tag{3}$$

where ε is an arbitrarily chosen maximal acceptable error.

Multi-parameter large-change sensitivity can be calculated recursively using the two-port transimpedance approach [3,4]:

$$Z_\xi(\boldsymbol{\alpha},\boldsymbol{\beta}) = Z_{\xi-1}(\boldsymbol{\alpha},\boldsymbol{\beta}) + K_{\xi-1}Z_{\xi-1}(\boldsymbol{\alpha},\boldsymbol{\xi})Z_{\xi-1}(\boldsymbol{\xi},\boldsymbol{\beta}), \tag{4a}$$

where: $\xi = 1,2,\ldots,m$,

$\boldsymbol{\xi} = (\xi_1,\xi_2)$ – pair of nodes to which the ξ-th element is connected,

$Z_0(\boldsymbol{\alpha},\boldsymbol{\beta})$ – nominal transimpedance between ports $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$,

$Z_\xi(\boldsymbol{\alpha},\boldsymbol{\beta})$ – transimpedance after change of ξ-th parameter,

$$K_{\xi-1} = -\frac{1}{1/\Delta y_\xi + Z_{\xi-1}(\boldsymbol{\xi},\boldsymbol{\xi})}. \tag{4b}$$

To calculate changes in the network function it is sufficient to know the changes in appropriate transimpedances. When parameters change by 100%, a situation that occurs with removal of elements, the admittance variation equals: $\Delta y_\xi = -y_\xi$. In such case the coefficient $K_{\xi-1}$ in (4a) is equal to:

$$K_{\xi-1} = \frac{y_\xi}{1 - y_\xi Z_{\xi-1}(\boldsymbol{\xi},\boldsymbol{\xi})}. \tag{5}$$

If an element is removed by short-circuiting, then $\Delta y_\xi \rightarrow \infty$ and

$$K_{\xi-1} = -\frac{1}{Z_{\xi-1}(\boldsymbol{\xi},\boldsymbol{\xi})}. \tag{6}$$

It is remarkable that the recursion (4a) works equally well for any arbitrary parameter change, including short- and open-circuiting of circuit components. Another very important feature of (4a) is the fact that it uses the transimpedances of the nominal (unperturbed) circuit. Each nominal transimpedance is calculated as a simple linear combination of at most four elements of the inverse of the node admittance matrix [3]. So, ultimately, application of the recursive formula (4a) requires a single numerical inversion of the node admittance matrix.

Our ability to calculate the multi-parameter extreme-change sensitivities means that we don't have to perform element-by-element elimination (each time recalculating the network function), but the entire set of candidate elements can be eliminated at once.

The choice of the elements to be eliminated is an NP-complete combinatorial problem. Fortunately, a simple heuristic can dramatically reduce the solution space. First, for each circuit element we calculate the single-parameter extreme sensitivities to s-c and o-c. The table of sensitivities is then sorted in ascending order. Next, the elements are chosen successively from this table and the multi-parameter extreme-change sensitivities for each set are calculated. The process continues until the predetermined error value is reached. There are several possible variations:

- the elements are chosen from the ordered list regardless of the sensitivity type (s-c or o-c),
- the elements with low open-circuit sensitivities are preferred,
- the elements with low short-circuit sensitivity are preferred.

Our experiments show that good results are obtained when as many as possible elements are eliminated by short-circuiting; then we try to remove elements with lowest open-circuit sensitivity.

3 ALGORITHM

The following algorithm results in a simplified formula for a network function; the simplified formula will produce numerical results with predetermined accuracy at given frequency points.

1. Determine a set of discrete frequencies $F = \{f_i: i = 1,2,\ldots,n\}$ and the acceptable variation ε of a network function at those frequencies:
$$\max_{f\in F}\left|\Delta T/T_0\right| < \varepsilon. \tag{7}$$
2. With nominal parameters, calculate the nodal admittance matrix $\mathbf{Y}$, its inverse $\mathbf{Z}$ and the network function at F.
3. Obtain single-parameter open- and short-circuit sensitivities for each parameter in $\mathbf{p} = [p_1, p_2, \ldots, p_m]^T$ according to:
$$\boldsymbol{\delta}_p = \max_{f\in F}\left|\Delta T/T_0\right|. \tag{8}$$
4. Sort the sensitivities in ascending order.
5. Until $\delta_{\mathbf{p}_j} \geq \varepsilon$, sequentially calculate the multi-parameter extreme-change sensitivities for the sub-vectors: $\mathbf{p}_j = [p_1, p_2,\ldots, p_j]^T$
$$\delta_{\mathbf{p}_j} = (\Delta T/T_0)\big|_{\Delta\mathbf{p}_j};\ j = 1,2,\ldots,m. \tag{9}$$

6. Repeat until condition (7) is satisfied:
   - $j = j - 1$;
   - remove the first j components from the circuit;
   - calculate the required network function of the modified circuit.

Step 6 is very often performed only once.

4 CIRCUIT EXAMPLE

To illustrate the proposed algorithm we will present a small circuit example [7].

Consider a BJT amplifier, shown in Fig. 1. The hybrid-π model parameters of the BJTs are: r_{x1} = 270, $r_{\pi 1}$ = 200k, g_{m1} = 1 mS, r_{o1} = 2.3M, $C_{\pi 1}$ = 20 pF, $C_{\mu 1}$ = 3 pF, r_{x2} = 270, $r_{\pi 2}$ = 4.5k, g_{m2} = 75 mS, r_{o2} = 35k, $C_{\pi 2}$ = 60 pF, $C_{\mu 2}$ = 3 pF. Our goal is to obtain a simplified symbolic expression for the voltage transmittance $T_v(s,\mathbf{p})$ of this amplifier. The expression should give at least 10% accuracy (ε = 0.1) of the magnitude of $T_v(s,\mathbf{p})$ in the frequency range 1kHz – 1MHz. As the function in question is quite smooth, we can select only a limited number of frequency points: F = {1k, 10k, 100k, 1M}.

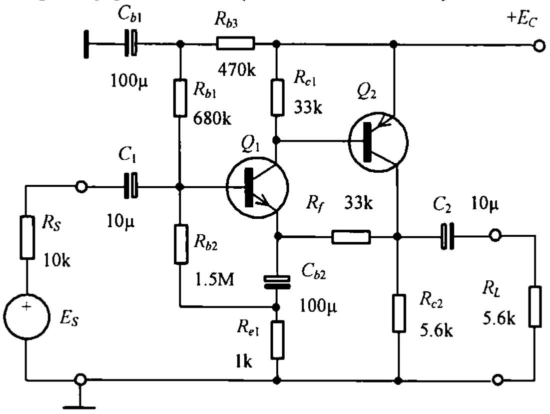

Fig. 1: A simple BJT amplifier.

After performing steps 2 – 4 of the algorithm in Section 3, we obtain a list of open- and short-circuit sensitivities. The relevant part of this list is shown in Table 1.

In step 5 we first calculate the multi-parameter short-circuit sensitivities for $\mathbf{p}_2$ = [R_{b3}, C_{b1}], $\mathbf{p}_3$ = [R_{b3}, C_{b1}, C_1], $\mathbf{p}_4$ = [R_{b3}, C_{b1}, C_1, C_{b2}], etc., until $\delta_{\mathbf{p}_j} \geq \varepsilon$. In our case $\delta_{\mathbf{p}_8} = -0.97 > \varepsilon$ and $\delta_{\mathbf{p}_7} = 0.0007 < \varepsilon$. Only seven components can be removed by short-circuiting (marked in **bold** in Table 1). Since the error is still much smaller than the allowed 10%, we keep adding more parameters to $\mathbf{p}$, taking them now from the list in ascending order of the open-circuit sensitivity (marked in *italics* in Table 1; note that the elements which have been already removed by s-c, are not included). Now we reach the point where $\delta_{\mathbf{p}_{14}} = 0.154 > \varepsilon$ and $\delta_{\mathbf{p}_{13}} = 0.073 < \varepsilon$. The iteration process is illustrated in Fig. 2.

Step 6 requires only one iteration to confirm that the accuracy condition is indeed satisfied.

So, to keep the relative error of $|T_v(s,\mathbf{p})|$ below 10%, we can eliminate 13 components from the equivalent small-signal circuit of the amplifier. This reduces the number of symbolic parameters in the voltage transmittance formula to 12.

Table 1: Ordered extreme-change sensitivities $|\Delta T_v/T_{v0}|$ for the amplifier in Fig. 1.

Rb3	o/c	1.145815e-013
Rb3	**s/c**	**3.383702e-008**
Cb1	**s/c**	**3.383702e-008**
C1	**s/c**	**2.693101e-005**
Ro1	*o/c*	*6.770211e-004*
Cb2	**s/c**	**1.484019e-003**
C2	**s/c**	**2.733563e-003**
Rb2	*o/c*	*3.050170e-003*
Cb1	o/c	5.982624e-003
Ro2	*o/c*	*1.110177e-002*
Rc1	*o/c*	*1.304126e-002*
Rb1	*o/c*	*1.476608e-002*
Rx1	**s/c**	**1.968737e-002**
Rp1	*o/c*	*2.369130e-002*
Rx2	**s/c**	**3.132154e-002**
Rc2	o/c	7.367996e-002
Rp2	o/c	8.496395e-002
Cp2	o/c	1.190218e-001
Cu1	o/c	3.038716e-001
Cp1	o/c	5.498861e-001
Re1	o/c	9.691058e-001
Cb2	o/c	9.693058e-001
Rf	s/c	9.975447e-001
Rc1	s/c	9.991137e-001
Rp2	s/c	9.993755e-001
		...

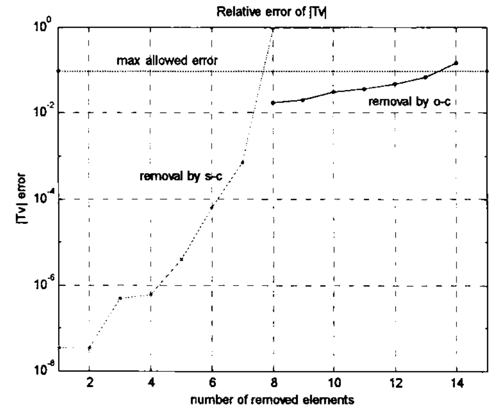

Fig. 2: Error behaviour during elimination of circuit elements.

It is important to emphasize the fact that to achieve this result we only need to calculate the inverse of the node admittance matrix of the original circuit once. All subsequent calculations are performed on this matrix using transimpedancies and the simple recursive formula (4a) in conjunction with (4b). Other methods require recalculation of the entire circuit after each modification.

With the modified circuit, the transmittance formula can be generated by any convenient method. For example, using our software STAINS [6], we obtain the sequential formula for the voltage transmittance:

```
d1  = (Gc2+Gf+GL+s*Cu2)/(Gm2-s*Cu2);
x1  = s*Cu1*d1;
x2  = -s*Cu2-(Gp2+s*(Cu1+Cp2+Cu2))*d1;
d2  = Gf/(s*Cu2-Gm2);
x3  = s*Cu1*d2-s*Cp1;
x4  = -Gm1-(Gp2+s*(Cu1+Cp2+Cu2))*d2;
d3  = x2/(x4);
x5  = x1-x3*d3;
x6  = -Gf-(Ge1+Gf+Gm1+s*Cp1)*d3;
d4  = (Gm1-s*Cu1)/(x4);
x7  = Gs+s*(Cp1+Cu1)-x3*d4;
x8  = -Gm1-s*Cp1-(Ge1+Gf+Gm1+s*Cp1)*d4;
d5  = x6/(x8);
x9  = Gs*d5;
x10 = x5-x7*d5;
Tv  = Gs/x10;
```

Fig. 3 shows the comparison between the voltage transfer magnitude of the original and the simplified circuits.

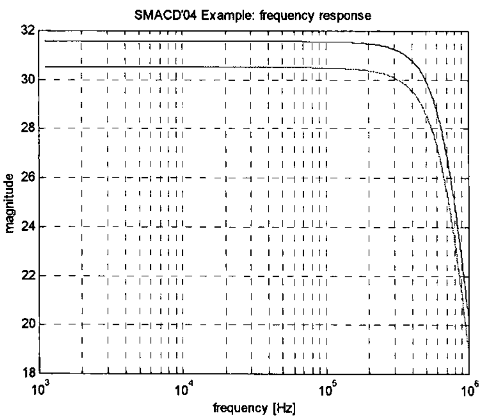

Fig. 3: Frequency response of the original (lower curve) and the simplified (upper curve) circuits.

5 CONCLUSION

We have presented a new method for simplification of symbolic network functions of linear electronic circuits. Our method is based on the application of the transimpedance concept [3] to calculation of the single-parameter and multi-parameter large-change sensitivities of network function to short- and open-circuiting of elements.

The entire process of selecting components for elimination requires only a single analysis of the nominal circuit (involving formulation of the node admittance matrix **Y** and inverting it numerically to obtain matrix **Z**). Components are selected for elimination using the ranking list obtained by sorting on the single-parameter extreme-change sensitivity.

Instead of removing elements one by one, each time recalculating the entire modified circuit, we compute the multi-parameter extreme-change sensitivity with successively expanding parameter list until a predetermined error value is reached. This calculation is performed with a simple recursive formula (4a) and involves only operations on the elements of the once calculated **Z**. The recursion (4a) handles both open- and short-circuit elimination of elements.

Our approach is conceptually simple and elegant. It requires less computational effort than methods based on element-by-element elimination and is independent of the symbolic technique employed in deriving the transmittance formula.

References

[1] B. Rodanski, M. Hassoun, "Symbolic Analysis Methods," in *The Circuits and Filters Handbook*, 2nd ed., W.-K. Chen, Editor. Boca Raton: CRC Press, 2003, pp. 1263-1282.

[2] F.V. Fernandez *et al.*, *Symbolic Analysis Techniques – Applications to Design*. IEEE Press, N.Y. 1998.

[3] F. Balik, B. Rodanski, "Calculation of Symbolic Sensitivities for Large-Scale Circuits in the Sequence of Expressions Form via the Transimpednace Method," *Analog Intergrated Circuits and Signal Processing*, 40, 2004, pp. 265-276.

[4] F. Balik, "Calculation of Multiparameter Large-Change Symbolic Sensitivities via the Transimpedance Method." *Int. Conf. on Signals and Electronic Systems, ICSES'01*. Łódź, Poland, Sept. 2001, pp. 307-312.

[5] J. Vlach, K. Singhal, *Computer Methods for Circuit Analysis and Design*, 2nd ed. New York: Van Nostrand Reinhold, 1994.

[6] L.P. Huelsman, "STAINS - Symbolic two-port analysis via internal node suppression," *IEEE Circuits & Devices Magazine*, vol. 18, no. 2, Mar. 2002, pp. 3-6.

[7] F. Balik, "Selected Problems of Symbolic Circuit Analysis," in *Selected Problems of Modelling, Design and Measurement of Low and High Power Analogue Electronic Circuits*, A. Prałat, Editor. Raport ITA PWr, I-28 SPR-2004, Wrocław, 2004 (in Polish).